

REMARKS

In the Office Action, it was noted that claims 1-30 were presented for examination. In fact, claims 1-56 are presented for examination.

Claims 1-56 were rejected under 35 U.S.C. § 112, second paragraph, as being indefinite. In particular, the phrase "optional attribute" and the term "object" were said to not be defined in the disclosure.

With regards to the term "object", Applicants direct the Examiner's attention to page 1, lines 13-16 of the specification which states that, "... an object is a set of programming instructions that supports one or more methods and/or has one or more properties."

With regards to the phrase "optional attribute", this phrase has its ordinary meaning. It is an attribute that is not mandatory. Applicants note that in claim 1 this is made clear by the phrase "that the objects should but are not required to have". The meaning of this phrase can also be easily determined by those skilled in the art, especially after reading page 2, line 16-29, which reads:

"Second, when searching Component Categories, it is difficult to find objects with desirable but not mandatory attributes. Under Component Categories, each of the search criteria is considered mandatory. If an object does not meet the search criteria, it is not returned. Thus, optional criteria cannot be placed in a first search for objects.

Because of this, an object or application performing a search must do several search iterations to find objects with optional attributes. In particular, the searching application must perform a first search for required attributes, then perform separate searches for each optional attribute."

Since the terms "object" and "optional attribute" are clearly defined in the specification, claims 1-56 are definite under 35 U.S.C. § 112.

Claims 1-12, 14-23, 25-29, 31 and 35-56 were rejected under 35 U.S.C. § 102(e) as being anticipated by Beall et al. (U.S. Patent Number 6,032,145, hereinafter Beall).

Beall provides a system for searching databases. Under the Beall system, mandatory search terms are accepted from a user and then a sequence of search algorithms are performed on a database. The search algorithms include a proximity search, a word count search, and a fuzzy logic search. When the match is found, the search is terminated and the search results are provided to the user. The user is then allowed to provide further mandatory search criteria, such as a category, in order to further limit the search results. For example, when the user selects a category, only those search results that meet this mandatory category are returned in the refined search results.

Claims 1-16

Independent claim 1 is a method of searching for programming objects. The method includes receiving instructions to search for objects, where the instructions include at least one optional attribute that the objects should but are not required to have. Objects are located in the computing environment and then are ordered in a list of objects based on matches between the attributes of the located objects and at least one optional attribute.

Beall does not show or suggest the invention of claim 1, because it does not show searching for programming objects. Instead, Beall is searching database entries for text string matches. As noted in the background of the present invention, a programming object is a set of programming instructions. Beall does not show or suggest performing a search for programming objects.

In addition, Beall does not receive instructions to search for objects where the instructions include at least one optional attribute that the objects should but are not required to have. In the Office Action, column 7, lines 44-46 and block 504 of FIG. 5 were cited as showing instructions that include at least one optional attribute. Applicants respectfully dispute this assertion.

The cited section does not show receiving instructions with optional attributes. Instead, this section shows the user refining their search by providing more mandatory attributes. The attributes provided during the refinement stage, such as a category, are mandatory attributes in that all of the search results that are returned during refinement must have these additional attributes. Beall does not return entries that do not match these additional attributes. Thus, none of the attributes that are provided during refinement can be considered optional, since all entries that are returned during the refinement must include them.

In addition, Beall does not show a step of ordering the located objects in a list of objects based on matches between the attributes of the located objects and at least one optional attribute. In the Office Action, column 7, lines 50-52 and 60-64 and step 505 of FIG. 5 were cited as showing such ordering based on optional attributes.

However, the cited sections make no mention of ordering a list of returned entries and as such can not show or suggest ordering such returned entries based on an optional attribute.

Since Beall does not search for optional attributes and does not order located objects based on optional attributes, it does not show or suggest the

invention of claim 1. As such, claim 1 and claims 2-16, which depend therefrom, are patentable over Beall.

Claims 4 and 5

Claims 4 and 5 depend from claim 1 and includes a limitation wherein locating objects comprises searching for sets of object data beneath a registry key. In rejecting claim 4, the Examiner asserted that column 2, lines 50-54 of Beall show searching beneath a registered key. The cited section makes no reference to a registry or a registry key. Further, it makes no mention of searching beneath a registry key. As such, claims 4 and 5 are further patentable over Beall.

Claim 10

Claim 10 depends from claim 1 and includes a further limitation wherein locating objects includes locating objects of an object class that is dynamically made available. In the Office Action, column 2, lines 47-54 were cited as showing locating objects of an object class that is dynamically made available. However, the cited section makes no reference to object classes or to object classes that are dynamically made available. As such, Beall does not show or suggest the invention of claim 10.

Claims 17-25

Independent claim 17 is directed to a computer-readable medium having instructions for searching for sets of object attributes and comparing found object attributes to a required attribute. A reference to those object attributes that match the required attribute are placed in a list and the list is ordered based on a comparison between the object attributes and an optional attribute.

The invention of claim 17 is not shown or suggested in Beall. In particular, Beall does not order a list based on a comparison between an object attribute and an optional

attribute. Although Beall does allow refinement of a search, this refinement results in narrowed search results.

It does not result in an ordering of a list. As such, claims 17-25 are patentable over Beall.

Claims 26-32

Independent claim 26 provides a method for searching for computer programming objects. The method includes receiving a request on a local computer to search for a programming object based on at least one search attribute. Based on the request, a search is preformed for object attributes associated with object classes that have an accessibility that is subject to change.

Beall does not show or suggest the invention of claim 26. In particular, Beall does not discuss object classes that have an accessibility that is subject to change. As such, Beall is not capable of showing or suggesting that a search be performed for object attributes that are associated with object classes that have an accessibility that is subject to change. Claims 26-32 are therefore patentable over Beall.

Claims 33-40

Independent claim 33 is directed to a computer-readable medium having instructions for receiving an instruction to search for an object based on at least one search attribute and determining that a set of object attributes could be located outside of a static attribute storage location. A search is performed for the set of attributes outside of the static attribute storage location and a reference to an object is returned based on a set of attributes found outside of the static attribute storage location.

Beall does not show or suggest the invention of claim 33 because it does not show or suggest a step of determining that a set of object attributes could be located outside of a static attribute storage location. Under Beall, the search is performed in a database. There is no mention of a set of attributes that could be located outside of a static attribute storage location. As such, Beall does not search for a set of attributes outside of a static attribute storage location and does not return a reference to an object based on a set of attributes found outside of a static attribute storage location. Therefore, claims 33-40 are patentable over Beall.

Claims 41-45

Independent claim 41 is directed to a method of instantiating and initializing a programming object. The method includes selecting an object data set for an object from a plurality of object data sets for the object's class, each object data set including the same unique identifier for the object's class. An object is then instantiated based on the unique identifier and the object is initialized using at least one attribute in the selected object data set.

Beall does not show or suggest the invention of claim 41. In particular, it does not discuss selecting an object data set for an object wherein each object data set includes the same unique identifier for the object's class. Further, it does not show instantiating any objects based on a unique identifier or of initializing an object using at least one attribute in the selected object data set. In short, Beall does not show any of the limitations of claim 41. As such, claims 41-45 are patentable over Beall.

Claims 46-51

Independent claim 46 is directed to a computer-readable medium having a computer loadable data structure. The data structure includes a first set of object data for an object class wherein the first set of object data comprises a first entry containing a unique identifier for the object class and at least one attribute of the object class. A second set of object data for the object class is also present in the data structure, wherein the second set of object data comprises a second entry containing the same unique identifier for the object class as the first entry and at least one attribute of the object class that is different from the at least one attribute of the first set of object data.

Beall does not show suggest the invention of claim 46. In particular, Beall makes no mention of object classes or unique identifiers for an object class. As such, claims 46-51 are patentable over Beall.

Claims 52-54

Independent claim 52 is directed to a computer-readable medium having a computer loadable object token that comprises computer-executable instructions. The computer-executable instructions provide for setting the object token to point to a set of object data related to an object and for retrieving attributes from the set of object data. The object token instructions further provide for instantiating the object based on a unique identifier in the set of the object data.

Beall does not show or suggest the invention of claim 52 because it does not mention instantiating an object based on a unique identifier in a set of object data. As such, claims 52-54 are patentable over Beall.

Claims 55-56

Independent claim 55 is directed to a computer-readable medium having a computer loadable token enumerator that includes instructions for locating object attributes that are located outside of a static attribute storage location and instructions for providing a reference to the object attributes.

Beall does not show or suggest the invention of claim 55 because it does not show or suggest instructions for locating object attributes that are located outside of a static attribute storage location. Under Beall, a single database 14 is examined. Beall does not show or suggest that attributes can be located outside of this database. As such, Beall does not show or suggest locating object attributes outside of a static attribute storage location. Claims 55 and 56 are thus patentable over Beall.

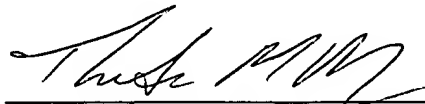
Conclusion

In light of the above remarks, claims 1-56 are patentable over Beall. Reconsideration and allowance of the claims is respectfully requested.

The Director is authorized to charge any fee deficiency required by this paper or credit any overpayment to Deposit Account No. 23-1123.

Respectfully submitted,

WESTMAN, CHAMPLIN & KELLY, P.A.

By: 
Theodore M. Magee, Reg. No. 39,758
Suite 1600 - International Centre
900 Second Avenue South
Minneapolis, Minnesota 55402-3319
Phone: (612) 334-3222 Fax: (612) 334-3312

TMM:sew